RTM Class and Attribute Definitions as of 06/25/96

This data dictionary provides definitions of classes and attributes in RTM database. It should be used with the RTM Class Definition Diagram to fully understand the relationship between classes. The RTM Class Definition Diagram describes the database architecture and it is posted on the ECS ccMail RTM bulletin board and ECS Requirements home page.

Forward any comments to Dat Vu, 301-925-0705, dvu@eos.hitc.com.

LEVEL_2

Contains requirements specified in Level 2 - Volume 1 and Volume 0. Objects in this class are mapped to objects in L3_FPRS class.

Owner of the class: **SMO**

Attribute Name	Attribute Description
source_req_id	**Mandatory**. Unique. Requirement identifier (i.e. 1234) 12 octets
text	**Mandatory**. Requirement text up to 16K octets
paragraph_id	**Mandatory**. Paragraph identifier, free text up to 60 octets
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created, replaced, or substituted. It is the index to the database and is also used to link objects from different classes.
source_requirement	Y/N. Indicates whether or not this requirement came from a source document.
current_status	Status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of the requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'

date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the requirement. Free text up to 16K octets.

L3_FPRS

Contains the Level 3, Functional and Performance Requirements Specifications received from GSFC 07/94 (423-41-02). Objects in this class can be mapped to objects in LEVEL_2, IRD, itself and REQ_BY_REL classes. All relationships are defined as many-to-many relationship.

Owner of the class: **SMO**

Attribute Name	Attribute Description
req_source_id	**Mandatory**. Unique. Requirement identifier i.e. EOSD1500, 15 octets
req_title	**Mandatory**. Short description of requirement, 35 octets
paragraph_id	**Mandatory**. Paragraph identifier, free text up to 60 octets
text	**Mandatory**. Requirement text up to 16K octets
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document.
current_status	Status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created, replaced, or substituted. It is used as index to the database and is also used to link objects from different classes.
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'

query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the requirement. Free text up to 16K octets.

REQ_BY_REL

Contains requirements allocated for each formal release. These requirements are expanded from L3_FPRS and IRD classes. It is used by development engineers to develop the Level 4 requirements. Objects in this class are mapped to objects in LEVEL_4, CCR, IRD, L3_FPRS, AT_A, AT_B, IT_Ir1, IT_A, IT_B, and IT_FOS classes. All relationships are defined as many-to-many relationship.

Owners of the class: **SMO; IATO; SCDO**.

Attribute Name	Attribute Description
paragraph_id	**Mandatory**. **Unique**. Requirement identifier expanded from L3_FPRS or IRD class plus the release identifier. The requirement id and release are separated by a "#" sign. Example: **EOSD1500#Ir1** **EOSD1500#A** **EOSD1500#B**
segment_allocation	**Mandatory**. Describes segment(s) allocated to the requirement. Select one or more from the list below: **'CSMS'** **'FOS'** **'SDPS'**
req_type	**Mandatory**. Indicates type of requirement. Select one or more from the list below: **'performance'** **'functional'** **'operational'** **'procedural'** **'interface'** **'security'** **'RMA'** **'standards'** **'evolvable'** **"procedural"** The requirement type "procedural" defined in the REQ_BY_REL class is used to describe requirements that are not required to have links to the lower level requirements. Examples of requirements that fall into this category are: requirements that are documenting a statement of work (SOW) activity; requirements that come from

	an IRD but are representing the other side of the interface; requirements that are referring to other documents; etc. Therefore, all "procedural" type requirements in the REQ_BY_REL class will not have associated children (i.e. Level_4, external_interface) mapped to them. All "procedural" type requirement allocations and reallocations must be approved by the ECS CCB before they are implemented in the RTM. **"operational"** The requirement type "operational" defined in the REQ_BY_REL class is used to describe requirements that require human (operator) action as a mandatory/necessary major component for satisfying the requirment.
req_category	**Mandatory**. Indicates criticality of the requirement. Select one from the list below: **'mission critical'** **'mission essential'** **'mission fulfillment'** **"mission critical"** No data loss, ensure system does not lose any lower level data needed to generate downstream data (e.g., higher level products). All real-time command/ control/telemetry monitoring requirements must be considered critical. **"mission essential"** Basic services for the long term data storage, Data management necessary to provide services to the user community that serves the majority of earth science researcher service needs in basic data distribution. **"mission fulfillment"** Advanced services specifically targeted at increasing the earth science user's productivity. Advanced services specifically targeted at meeting larger programmatic goals. Advanced services providing intermediary support of educational, policy, social services community Advances services for access to GCDIS and UserDIS.
s_verification_method	**Mandatory**. Indicates how the requirement will be verified by SI&T. Select one from the list below: **'test'** **'demo'** **'analysis'**

	'inspection'
s_verification_status	**Mandatory**. Indicates requirement verification status by SI&T. Select one from the list below: **'verified'** **'partially verified'** **'un-verified'** Default option is 'un-verified'
a_verification_method	**Mandatory**. Indicates how the requirement will be verified by IATO. Select one from the list below: **'test'** **'demo'** **'analysis'** **'inspection'** **'N/A_Ir1'**
a_verification_status	**Mandatory**. Indicates requirement verification status by ATO. Select one from the list below: **'verified'** **'partially verified'** **'un-verified'** **'N/A_Ir1'** Default option is 'un-verified'
text	**Mandatory**. Requirement text up to 16K octets
req_interpretation	Text interpretation of requirement, free text up to 16K octets.
release	Indicates the release where the requirement will be delivered. Select one from the list below: **'Ir1'** **'A'** **'B'** **'C'** **'D'** **'TK5'** **'TK5a'** **'TK5b'**
req_title	Short description of requirement, 35 octets
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created, replaced, or substituted. It is used as index to the database and is also used to link objects from different classes.

source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document. If not (N), it is a derived requirement.
current_status	Status of requirement updated and maintained by RTM. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets.
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text which further clarifying the complexity of the requirement. Free text up to 16K octets.

LEVEL_4

Contains Level 4 requirements which were expanded from the REQ_BY_REL class. Objects in this class are mapped to objects in REQ_BY_REL, IT_FOS, IT_A, IT_Ir1, IT_B, COTS, CCR, and COMPONENT classes. All relationships are defined as many-to-many relationship.

Owners of the class: **SCDO; FOS**

Attribute Name	Attribute Description
paragraph_id	**Mandatory**. **Unique**. Requirement id, free text up to 60 octets. Format: [seg-id]-[subsystem-id]-[seq#] [seg-id] 1 character identifies an ECS segment C for CSMS F for FOS S for SDPS [subsystem-id] a 3-character uniquely identifies a subsystem within ECS: CSMS: MSS, ISS, CSS FOS: PAS, CMS, CMD, TLM, RMS, ANA, FUI, DMS, FOS, HRD SDPS: DMS, PLS, CLS, IOS, INS, DSS, DPS Toolkit: TKD, TKS (Note: TKD and TKS toolkit IDs are not subsystems per se, but subsystem indicators used to denote the DAAC toolkit version and the SCF toolkit version, respectively. The TKD L4 requirements will map to the standard Ir1, A, B and C release fields whereas the TKS L4 requirements will map to TK5, TK5a and TK5b release fields only) [seq#] 5-digit sequential number
req_type	**Mandatory**. Indicates the type of requirement. Select one or more from the list below: **'performance'** **'functional'** **'operational'** **'procedural'** **'interface'** **'security'** **'RMA'** **'standards'** **'evolvable'**

release	**Mandatory**. Indicates the release that in which the requirement will be delivered. Select one or more from the list below: **'A'** **'B'** **'C'** **'D'** **'Ir1'** **'EP1'** **'EP2'** **'EP3'** **'EP4'** **'EP5'** **'EP6'** **'TK3'** **'TK4'** **'TK5'** **'TK5a'** **'TK5b'**
req_status	**Mandatory**. Indicates status of the requirement. Select one from the list below: **'approved'** **'disapproved'** **'in-review'** **'delayed'** **'agreed'** **'incremental'**
verification_method	**Mandatory**. Indicates method of verification. Select one from the list below: **'test'** **'demo'** **'analysis'** **'inspection'** **'test/demo'** **'inspection/test'** **'test/analysis'** **'demo/inspection'**
text	**Mandatory**. Requirement text up to 16K octets.
section_number	List the DID 304 section number that contains this requirement. Alpha-numeric, 12 octets.
source_interface	Source identifier of the interface. Must be entered if requirement type is the interface requirement, 15 octets.

destination_interface	Destination identifier of the interface**.** Must be entered if requirement type is the interface requirement, 15 octets.
relb_sec_no	Section number of the requirement. Alpha-numeric, 12 octets.
reqt_comment	Additional comment which further clarifying the requirement. Alpha-numeric, 75 octets.
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document. If not (N), it is a derived requirement.
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created, replaced, or substituted. It is used as index to the database and is also used to link between objects from different classes.
current_status	System status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the requirement. Free text up to 16K octets.

COTS

Contains the commercial-off-the-shelf (COTS) software packages that are being used to develop software for ECS. Objects in this class are mapped to objects in LEVEL_4 class. Relationship between two classes is defined as many-to-many relationship.

Owners of the class: **SCDO, FOS**

Attribute Name	Attribute Description
COTS_ID	**Mandatory**. **Unique**. Identifies COTS name. 25 octets. Example: "**Autosys**"
COTS_desc	**Mandatory**. Text description of COTS. 75 octets.
vendor_name	**Mandatory**. Indicates COTS vendor information (name, phone number). 75 octets.
object_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created, replaced, or substituted. It is used as index to the database and is also used to link between objects from different classes.
lifecycle_status	System status of object. **'current'** **'replaced'** **'substituted'** **'deleted'**

AT_A

Contains the system acceptance test sequences and test cases for A as identified in Acceptance Test Plan (ATP) and Acceptance Test Procedures. Objects in this class are mapped to objects in REQ_BY_REL class. Relationship between two classes is defined as many-to-many relationship.

Owner of the class: **IATO**

Attribute Name	Attribute Description
test_number	**Mandatory**. **Unique**. Identifies the test procedure number. A combination of the release, group, scenario, sequence, test case number and site ID used to verify each requirement in the REQ_BY_REL class. 13 octets. Format: RGGssSS.XXX$S where: R release identifier (A,B,C,D) GG group number as identified in ATP 08 - system Management 09 - push 10 - pull 11 - FOS 12 - end-to-end ss scenario number (00-90) as identified in ATP SS sequence number (00-99) incr by 10 as identified in ATP .XXX test case number (000-999) incr by 10 as identified in ATP $ delimiter S a character identifying test site 'E'- EDC DAAC 'F'- EOC 'G'- GSFC DAAC 'L'- LaRC DAAC 'M'- MSFC DAAC 'S'- SMC Example: "A080210.110$E" - test number A080210.110 conducted at EDC DAAC.
seq_title	**Mandatory**. Identification name of test sequence from the plan/procedure document. Free text up to 40 octets
test_title	**Mandatory**. Identification name of the test procedure. Free text up to 40 octets

site_id	**Mandatory**. Identify site(s) that test is applicable to. Select one or more from list below: **'SMC'** **'EOC'** **'GSFC'** **'MSFC'** **'LaRC'** **'EDC'**
procedure_paragraph_id	**Mandatory**. Identifies the location within the AcceptanceTest Plan (DID409) where a detailed description of the test case is found. 12 octets
method_verification	**Mandatory**. Indicates method of verification. Select one from the list below: **'test'** **'demo'** **'analysis'** **'inspection'**
test_status	**Mandatory**. Identifies the current level of testing. Select one from list below: **'incomplete'** **'dry-run'** **'lien'** **'complete'** Default option is 'incomplete'.
remarks	Further clarification about this unique test sequence or test case. Free text up to 16K octets.
object_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created, replaced, or substituted. It is used as index to the database and is also used to link between objects from different classes.
lifecycle_status	System status of object. **'current'** **'replaced'** **'substituted'** **'deleted'**

AT_B

Contains the system acceptance test sequences and test cases for B as identified in Acceptance Test Plan (ATP) and Acceptance Test Procedures. Objects in this class are mapped to objects in REQ_BY_REL class. Relationship between two classes is defined as many-to-many relationship.

Owner of the class: **IATO**

Attribute Name	**Attribute Description**
test_number	**Mandatory**. **Unique**. Identifies the test procedure number.A combination of the release, group, scenario, sequence, test case number and site ID used to verify each requirement in the REQ_BY_REL class. 13 octets. Format: RGGssSS.XXX$S where: R release identifier (A,B,C,D) GG group number as identified in ATP 08 - system Management 09 - push 10 - pull 11 - FOS 12 - end-to-end ss scenario number (00-90) as identified in ATP SS sequence number (00-99) incr by 10 as identified in ATP .XXX test case number (000-999) incr by 10 as identified in ATP $ delimiter S a character identifying test site 'A' ASF DAAC 'E' EDC DAAC 'F' EOC 'G' GSFC DAAC 'J' JPL DAAC 'L' LaRC DAAC 'M' MSFC DAAC 'N' NSIDC DAAC 'O' ORNL DAAC 'S' SMC Example: "A080210.110$E" - test number A080210.110 conducted at EDC DAAC.
seq_title	**Mandatory**. Identification name of test sequence from the plan/procedure document. Free text up to 40 octets

test_title	**Mandatory**. Identification name of the test procedure. Free text up to 40 octets
site_id	**Mandatory**. Identify site(s) that test is applicable to. Select one or more from list below: **'SMC'** **'EOC'** **'GSFC'** **'MSFC'** **'LaRC'** **'EDC'** **'NSIDC'** **'ASF'** **'JPL'** **'ORNL'**
procedure_paragraph_id	**Mandatory**. Identifies the location within the AcceptanceTest Plan (DID409) where a detailed description of the test case is found. 12 octets
method_verification	**Mandatory**. Indicates method of verification. Select one or more from the list below: **'test'** **'demo'** **'analysis'** **'inspection'**
test_status	**Mandatory**. Identifies the current level of testing. Select one from list below: **'incomplete'** **'dry-run'** **'lien'** **'complete'** Default option is 'incomplete'.
text	**Mandatory**. Requirement text up to 16K octets
remarks	Further clarification about this unique test sequence or test case. Free text up to 16K octets.
requirement_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created, replaced, or substituted. It is used as index to the database and is also used to link between objects from different classes.
paragraph_id	Paragraph identifier, free text up to 60 octets

source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document (i.e. was auto_stripped)
current_status	System status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. Used to rebuild the requirement document.
date_created	Requirement creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the requirement. Free text up to 16K octets.

IT_A

Contains segment's and system's build/thread test cases used to verify each release A requirement in REQ_BY_REL class and LEVEL_4 class. Objects in this class are mapped to objects in REQ_BY_REL class and objects in LEVEL_4 class. Relationships between classes are defined as many-to-many relationships.

Owner of the class: **SCDO-I&T**

Attribute Name	**Attribute Description**
test_id	**Mandatory**. **Unique**. Test identifier, alpha-numeric 12 octets. Format: TSnnn.ccc where: T Thread or build identifier T for thread or B for build S segment identifier C for CSMS S for SDPS Y for System nnn thread or build number (000-999) .ccc test case number (000-999) Example: "TC001.001" is a thread, belonging to CSMS, with thread number 001 and test case number 001
test_name	**Mandatory**. Name of thread or build, free text up to 75 octets
seg_id	**Mandatory**. Segment identifier. Select one or more from list below: **'CSMS'** **'SDPS'**
test_case_name	**Mandatory**. Name of test case, free text up to 75 octets
verification_method	**Mandatory**. Indicates how the requirement will be verified. Select one from the list below: **'test'** **'demo'** **'analysis'** **'inspection'**
test_status	**Mandatory**. Status of the test. Select one from list below:

	'verified' **'unverified'** **'partially verified'** Default is 'unverified'.
text	**Mandatory**. Text description of test case, free text up to 16K octets
success_criteria	Defines the criteria that must be met to successfully complete the test case, free text up to 16K octets
misc_trace	Text description describing: - mapping to other obsolete tests due to the consolidation - mapping of objectives to tests (e.g. prototypes in the release) - mapping of ICD sections to tests - other Free text up to 16K octets
paragraph_id	Paragraph identifier, free text up to 60 octets
requirement_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created, replaced, or substituted. The key is also used to link between objects from different classes.
current_status	System status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document (i.e. was auto_stripped)
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'

query	Additional text describing uniqueness of test. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the test. Free text up to 16K octets.

IT_B

Contains segment's and system's build/thread test cases used to verify each release B requirement in REQ_BY_REL class and LEVEL_4 class. Objects in this class are mapped to objects in REQ_BY_REL class and objects in LEVEL_4 class. Relationships between classes are defined as many-to-many relationships.

Owner of the class: **SCDO-I&T**

Attribute Name	**Attribute Description**
paragraph_id	**Mandatory**. **Unique**. Test identifier, numeric 12 octets. Format: TSnnn.ccc where: T Thread or build identifier T for thread or B for build S segment identifier C for CSMS S for SDPS Y for System nnn thread or build number (000-999) .ccc test case number (000-999) Example: "TC001.001" is a thread, belonging to CSMS, with thread number 001 and test case number 001
test_name	**Mandatory**. Name of thread or build, free text up to 75 octets
seg_id	**Mandatory**. Segment identifier. Select one from list below: **'CSMS'** **'SDPS'**
test_case_name	**Mandatory**. Name of test case, free text up to 75 octets
verification_method	**Mandatory**. Indicates how the requirement will be verified. Select one from the list below: **'test'** **'demo'** **'analysis'** **'inspection'**
test_status	**Mandatory**. Status of the test. Select one from list below:

	'verified' **'unverified'** **'partially verified'** Default is ‘unverified’.
text	**Mandatory**. Text description of test case, free text up to 16K octets
success_criteria	Defines the criteria that must be met to successfully complete the test case, free text up to 16K octets
misc_trace	Text description describing: - mapping to other obsolete tests due to the consolidation - mapping of objectives to tests (e.g. prototypes in the release) - mapping of ICD sections to tests - other Free text up to 16K octets
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document (i.e. was auto_stripped)
requirement_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created, replaced, or substituted. The key is also used to link between objects from different classes.
current_status	System status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of test. Free text up to 16K octets.

clarification	Additional text further clarifying the complexity of the test. Free text up to 16K octets.

IT_FOS

Contains segment's and system's build/thread test cases used to verify each FOS requirement in REQ_BY_REL class and LEVEL_4 class. Objects in this class are mapped to objects in REQ_BY_REL class and objects in LEVEL_4 class. Relationships between classes are defined as many-to-many relationships.

Owner of the class: **FOS-I&T**

Attribute Name	Attribute Description
paragraph_id	**Mandatory**. **Unique**. Test id, free text up to 60 octets
text	**Mandatory**. Text description of the test, free text up to 16K octets
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document (i.e. was auto_stripped)
requirement_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created, replaced, or substituted. The key is also used to link between objects from different classes.
current_status	System status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of test. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the test. Free text up to 16K octets.

EXTERNAL_INTERFACE Contains descriptions of all external interfaces that ECS interacts with. Objects in this class are mapped to objects in REQ_BY_REL class and COMPONENT class. All relationships between classes are defined as many-to-many relationship.

Owner of the class: **Interface Group**

Attribute Name	Attribute Description
ext_interface_id	**Mandatory**. Unique. Name identifier for the data flow in the ICD. This field contains the External I/F name, e.g. TSDIS and the interface flow, e.g. DAN. A "#" sign separates the two , e.g., TSDIS#DAN. Up to 75 octets.
interface_verification_status	**Mandatory**. Indicates consistency status between design documents (DID305, DID313) and the ICD. Select one from list below: **'verified'** design document is consistent with ICD **'inconsistent'** design document is not consistent with ICD **'incomplete'** inadequate information to conduct verification Default: 'incomplete'
comment	Additional comment which further clarifying the interface. Alpha-numeric, 75 octets.
object_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created or substituted. It is used as index to the database and is also used to link between objects from different classes.
lifecycle_status	Status of object. **'current'** **'replaced'** **'substituted'** **'deleted'**

COMPONENT

Contains the characteristics of CSCIs, components (CSC/HWC), and OMT objects that are being developed. Objects in this class are mapped to objects in LEVEL_4 class and EXTERNAL_INTERFACE class. All relationships between classes are defined as many-to-many relationship.

Owners of the class: **SCDO, FOS**

Attribute Name	Attribute Description
paragraph_id	**Mandatory**. **Unique**. Contains name of the component. 60 octets.
component_type	**Mandatory**. Indicates this component is CSCI, CSC, HWC, or OMT Object. Select one from the list below: **'Subsystem'** **'CI'** **'CSC'** **'OBJECT'**
dev_category	**Mandatory**. Indicates how the component will be developed. Select one or more from the list below: **'COTS'** **'Develop'** **'Modify COTS'**
text	**Mandatory**. Text description of the component indicated by the component_type attribute, up to 16K octets
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document (i.e. was auto_stripped)
requirement_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created or substituted. The key is also used to link between objects from different classes.
current_status	Status of object. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of object creator, 8 octets

document_id	Name of the source document where this requirement was stripped. Used to rebuild the requirement document.
date_created	Object creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of object. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the object. Free text up to 16K octets.

CCR

Contains Change Control Request (CCR) information that was authorized by the Change Control Board (CCB) to update a set of requirements in the requirement baseline. Objects in this class are mapped to objects in REQ_BY_REL, IRD, and LEVEL_4 classes. All relationships between classes are defined as many-to-many relationship.

Owner of the class: **CMO**

Attribute Name	Attribute Description
CCR_id	**Mandatory**. Unique. Contains the CCR number generated and authorized by the CCB. Alpha-numeric 20 octets.
CCR_revision	**Mandatory**. Contains the CCR revision number generated and authorized by the CCB. Alpha-numeric 5 octets.
originator	**Mandatory**. Contains the name of person who originated the CCR. Alpha-numeric 30 octets.
board_name	**Mandatory**. Contains the name of CCB board. Alpha-numeric 30 octets.
approval_date	**Mandatory**. Contains the date the CCR was approved. Alpha-numeric 11 octets. Format: DD-MMM-YYYY
entry_status	**Mandatory**. Used to track status of CCR entry of requirement changes. Values are: **'implementation in process'** designates CCR entering the CCB process, but has not yet started or completed data entry **'input complete'** indicates requirements update in RTM has been completed but not yet verified by Quality Office **'validation complete'** indicates that the change authorized by the CCR is validated and completed in RTM Default: 'implementation in process'.
text	**Mandatory**. Text description of the CCR, up to 16K octets
paragraph_id	Paragraph identifier, free text up to 60 octets

source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document
requirement_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM whenever an object is created or substituted. The key is also used to link between objects from different classes.
current_status	Status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source document where this object was stripped. Used to rebuild the source document.
date_created	CCR creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of object. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the object. Free text up to 16K octets.

IRD

Contains external interface requirements specified in Interface Requirements Documents (IRDs). Objects in this class are mapped to objects in L3_FPRS, CCR, and REQ_BY_REL classes. All relationships between classes are defined as many-to-many relationship.

Owner of the class: **Interface Group**

Attribute Name	Attribute Description
req_source_id	**Mandatory**. **Unique**. Requirement identifier i.e. EOSD1500, 15 octets
source_interface	**Mandatory**. Specifies the name where the interface originates, 12 octets
destination_interface	**Mandatory**. Specifies the name where the interface ends, 12 octets
text	**Mandatory**. Requirement text up to 16K octets
paragraph_id	Paragraph identifier, free text up to 60 octets
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document (i.e. was auto_stripped)
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created or substituted. It is used as index to the database and is also used to link between objects from different classes.
current_status	System status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. Used to rebuild the requirement document.
date_created	Requirement creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'

date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the requirement. Free text up to 16K octets.

PROTOTYPE

Contains identified ECS prototypes and studies. Objects in this class are mapped with objects in RISK and TECH_ASSESS classes. Relationships between classed are defined as many-to-many relationship.

Attribute Name	Attribute Description
id	**Mandatory**. Unique. Numeric key of the prototype/study, alpha-numeric, 5 octets. Format: P or S followed by 4 digits, i.e. P0001 for prototype 1, S0001 for study 1.
type	**Mandatory**. Indicates this is a prototype or study. Select one from the list below: **'prototype'** **'study'**
title	**Mandatory**. Title of the prototype/study, alpha-numeric, 45 octets.
category	**Mandatory**. Prototype/study category. Alpha-numeric 20 octets. Suggested values: **'advanced'** **'engineering'** **'technology'**
purpose	**Mandatory**. Purpose of the prototype/study. Select one from below: **'requirement clarification'** **'risk mitigation'** **'functional enhancement'** **'technology assessment'**
release	**Mandatory**. Identify associated release with the prototype/study. Select one or more from the list below: **'Ir1'** **'Release A'** **'Release B'** **'Release C'** **'Release D'**
sub_system	Identify associated subsystem with the prototype/study. Select one from below **'F-ANA'** **'F-CMD'** **'F-CMS'** **'F-DMS'** **'F-FUI'** **'F-PAS'**

'F-RMS'
'F-TLM'
'C-CSS'
'C-ISS'
'C-MSS'
'C-HRD'
'S-CLS'
'S-DMS'
'S-DPS'
'S-DSS'
'S-INS'
'S-IOS'
'S-PLS'

principal_investigator	Name of the principal investigator, 20 octets
lead_by	Name of prototype/study lead, 20 octets
lead_address	Contains phone number and/or e-mail address of the prototype lead, 75 octets.
ESDIS_eval_leader	Name of the ESDIS counterpart evaluation team leader, 20 octets
eval_team_leader	Name of evaluation team leader, 20 octets
dev_team_rep	Name of the development team representative, 20 octets
effort	Estimated effort in man-month to develop the prototype/study, 3 bytes numeric
eval_method	Methods used for evaluation. Free text up to 16K octets
funding	A short statement on how the prototype/study be funded and should identify the funding source i.e. ECS, EDOS, ECOM, ESDIS, HQ/NRA, HQ/X, 60 octets
other_significant_cost	Clarification of additional cost to develop the prototype/study. Free text up to 16K octets
key_risk_addressed	List key risk areas associated with the prototype/study. Free text up to 16K octets
objective	Describe objective of the prototype/study. Free text up to 16K octets
approach	Describe approach to do the prototype/study, including study methodology, key resources

	consulted or required, tools to be used, candidate HW or SW, etc. Free text up to 16K octets
prior_segment	Identifies segment(s) associated with the prototype/study. Select one or more from the list below: **'FOS'** **'SDPS'** **'CSMS'** **'NONE'**
document	List associated documents. Free text up to 16K octets
key_prod	List of key products associated with the prototype/study. Free text, up to 16K octets
key_events	List of key events associated with the prototype/study, i.e. PDR, CDR, EP, PW. Free text, up to 16K octets
organization	Identifies associated organization with the prototype/study. Select one or more from the list below: **'FOS'** **'MRS'** **'SCDO'** **'SO'**
start_date	Date work initiated, 11 octets. Format: DD-MMM-YYYY
end_date	Date work completed, 11 octets. Format: DD-MMM-YYYY
monthly_update	Describe monthly status of the prototype/study. Entries start with month and year, i.e. July 1995 -
obj_status	Indicates prototype/study status. Select one from the list below **'proposed'** **'rejected'** **'in-progress'** **'completed'** **'integrated'**
results	Describe result and status of the prototype/study. Free text up to 16K octets
object_key	Internal object key to the database. This numeric sequence key is automatically generated by RTM

	whenever an object is created or substituted. It is used as index to the database and is also used to link between objects from different classes.
lifecycle_status	Status of object. **'current'** **'replaced'** **'substituted'** **'deleted'**

RISK

Contains information associated to the program risk activities. Objects in this class are mapped to objects in PROTOTYPE class and the TECH_ASSESS class where appropriate. All relationships between classes are defined as many-to-many relationship.

Attribute Name	Attribute Description
risk_id	**Mandatory.** An identifier for each risk item. Format: 1 letter extracted from the first letter of risk category below, a '-', and two digit number, i.e. **U-01** for user interaction risk # 1
risk_item	**Mandatory.** Short description of the risk, 60 octets
risk_category	**Mandatory**. Identify the category of the risk. Select one from the list below: **'Architecture'** **'Technology'** **'Evolution'** **'System Operations'** **'Programmatics'** **'User Interaction'**
resp_office	**Mandatory.** Name of the ECS office who is responsible for the risk. Select one from the list below: **'FOS'** **'CSMS'** **'SDPS'** **'SIP'** **'SCIENCE OFFICE'**
risk_factor	**Mandatory.** Contains the numeric factor of the risk, i.e. 0.62. Format x.xx. It is calculated from the probability of failure and the consequence of failure. Number ranging from 0.00 to 1.00.
text	Mandatory. Risk description text up to 16K octets
paragraph_id	Paragraph identifier, free text up to 60 octets
source_requirement	Y/N. Indicates whether or not this risk came from a risk document. If not (N), it is a derived risk.
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a risk is created or substituted. It is used as index to the database and is also used to link objects from different classes.

current_status	System status of risk. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of risk creator, 8 octets
document_id	Name of the source risk document where this risk was stripped. Used to rebuild the risk document.
date_created	Risk creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of risk. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the risk. Free text up to 16K octets.

TECH_ASSESS Contains information associated to the Technology Assessment activities.

Attribute Name	Attribute Description
TBD	

RTM Class and Attribute Definitions as of 06/25/96

This data dictionary provides definitions of classes and attributes in RTM database. It should be used with the RTM Class Definition Diagram to fully understand the relationship between classes. The RTM Class Definition Diagram describes the database architecture and it is posted on the ECS ccMail RTM bulletin board and ECS Requirements home page.

Forward any comments to Dat Vu, 301-925-0705, dvu@eos.hitc.com.

LEVEL_2

Contains requirements specified in Level 2 - Volume 1 and Volume 0. Objects in this class are mapped to objects in L3_FPRS class.

Owner of the class: **SMO**

Attribute Name	Attribute Description
source_req_id	**Mandatory**. Unique. Requirement identifier (i.e. 1234) 12 octets
text	**Mandatory**. Requirement text up to 16K octets
paragraph_id	**Mandatory**. Paragraph identifier, free text up to 60 octets
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created, replaced, or substituted. It is the index to the database and is also used to link objects from different classes.
source_requirement	Y/N. Indicates whether or not this requirement came from a source document.
current_status	Status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
original_editor	Logon id of the requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time. Format 'Wed Sep 8 14:52:43 1993'

date_last_modified	Date and time of last modification. Format 'Wed Sep 8 14:52:43 1993'
query	Additional text describing uniqueness of requirement. Free text up to 16K octets.
clarification	Additional text further clarifying the complexity of the requirement. Free text up to 16K octets.

L3_FPRS

Contains the Level 3, Functional and Performance Requirements Specifications received from GSFC 07/94 (423-41-02). Objects in this class can be mapped to objects in LEVEL_2, IRD, itself and REQ_BY_REL classes. All relationships are defined as many-to-many relationship.

Owner of the class: **SMO**

Attribute Name	Attribute Description
req_source_id	**Mandatory**. Unique. Requirement identifier i.e. EOSD1500, 15 octets
req_title	**Mandatory**. Short description of requirement, 35 octets
paragraph_id	**Mandatory**. Paragraph identifier, free text up to 60 octets
text	**Mandatory**. Requirement text up to 16K octets
source_requirement	Y/N. Indicates whether or not this requirement came from a requirement document.
current_status	Status of requirement. **'current'** **'replaced'** **'substituted'** **'deleted'**
requirement_key	Internal requirement key to the database. This numeric sequence key is automatically generated by RTM whenever a requirement is created, replaced, or substituted. It is used as index to the database and is also used to link objects from different classes.
original_editor	Logon id of requirement creator, 8 octets
document_id	Name of the source requirement document where this requirement was stripped. It is used to rebuild the source requirement document.
date_created	Requirement creation date and time , 25 octets. Format 'Wed Sep 8 14:52:43 1993'
date_last_modified	Date and time of last modification, 25 octets. Format 'Wed Sep 8 14:52:43 1993'